AMENDMENTS TO THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Claims 1-143 (withdrawn)

Claim 144 (original) A method of execution-instruction delegation between processing resources of at least two different types, comprising:

obtaining an execution instruction, wherein the execution instruction is obtained at [[a]] one of at least two processing resources of a first type;

determining whether an operation-code within the execution instruction should be delegated to an other processing resource of a second type different from the first type and shared by at least two processing resources of the first type;

executing the execution instruction with the processing resource of the first type, if the operation-code within the execution instruction should not be delegated to [[an]] the other processing resource; and

routing the execution instruction to [[an]] the other processing resource, if the operation-code within the execution instruction is for [[an]] the other processing resource.

Claim 145 (original) The method of claim 144, wherein the method is completed within a single processing cycle.

Claims 146-154 (withdrawn)

Claim 155 (original) The method of claim 144, wherein the operation-code indicates a type of resource on which to execute.

Claim 156 (currently amended) The method of claim 144, wherein ~~the other~~ at least one processing resource ~~may be the~~ of the first type is an originating processing resource.

Claim 157 (currently amended) The method of claim 144, wherein [[a]] the processing resource of the first type is an integer processing unit.

Claim 158. (currently amended) The method of claim 144, wherein [[a]] the processing resource of the second type is a mathematical processing unit.

Claim 159 (withdrawn)

Claim 160 (currently amended) The method of claim 144, wherein [[a]] the processing resource of the second type is a vector processing unit.

Claims 161-163 (withdrawn)

Claim 164 (currently amended) The method of claim 144, wherein [[a]] the processing resource of the second type is an execution-instruction processing cache.

Claim 165 (original) The method of claim 144, ~~wherein~~ further comprising routing ~~occurs~~ the execution instruction through an execution-instruction signal router.

Claim 166 (withdrawn)

Claim 167. (currently amended) The method of claim 144, wherein a first processing resource executing a first individual thread may sleep while ~~an other~~ a second processing resource executes delegated execution-instructions from the first individual thread.

Claim 168 (currently amended) The method of claim 144, wherein [[the]] an execution-instruction signal causes various processing resources dynamically to turn on and off to maintain a desired level of power draw while maximizing processing throughput.

Claim 169. (currently amended) The method of claim 144, further comprising generating an execution-instruction signal from processing resources, wherein the execution-instruction signal from the processing resources themselves shuts off processing resources while idling.

Claim 170 (currently amended) The method of claim 144, further comprising generating an execution-instruction signal from processing resources, wherein [[the]] an execution-instruction signal from processing resources themselves turn on processing resources when execution-instruction signal processing is required.

Claim 171. (original) The method of claim 144, wherein the processing resources are communicatively disposed on a same die.

Claim 172. (original) The method of claim 171, wherein an execution-instruction signal router is on the same die with processing resources.

Claims 173-1614 (withdrawn)

Claim 1615 (New) A method of execution-instruction delegation among multiple dependent processing resources, the processing resources comprising multiple elemental processing resources that are configured to perform processing operations according instructions in an instruction set and are individually incapable of servicing at least one instruction in the instruction set, the method comprising:

receiving one of multiple execution-instructions from a thread at a first Instruction Processing Unit (IPU) of at least two IPUs;

processing the first execution-instruction using the first IPU;

receiving an other execution-instruction from the thread at the first IPU;

determining that the other execution-instruction from the thread can not be processed by the first IPU;

delegating the other execution-instruction from the thread to a processing resource, of a type other than an IPU-type processing resource, that is shared between the first IPU and a second IPU, the type of the processing resource being such that it and can process the other execution-instruction from the thread; and

maintaining the thread in the first IPU in a sleep state until an indicator that the processing of the other execution-instruction from the thread by the processing resource is returned.

AMENDMENTS TO THE DRAWINGS

The drawings have been objected to because the text is illegible. Applicant submits herewith replacement sheets for Figures 1A, 1B, 3 and 4 that Applicant believes should remedy the above-noted objection to the drawings by providing larger, and hence more legible, text.